The 2009 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 11-15, 2009 St. Louis, USA

A Vision Based System for Attitude Estimation of UAVs

Saul Thurrowgood, Dean Soccol, Richard J. D. Moore, Daniel Bland, and Mandyam V. Srinivasan

Abstract—**This paper describes a technique for estimating the attitude of a UAV by monitoring the visual horizon. An algorithm is developed that makes the best use of color and intensity information in an image to determine the position and orientation of the horizon, and infer the aircraft's attitude. The technique is accurate, reliable, and fully capable of real-time operation. Furthermore, it can be incorporated into any existing vision system, irrespective of the way in which the environment is imaged (e.g. through lenses or mirrors).**

I. Introduction

Attitude stabilization is critical for any aircraft that is required to fly autonomously. Traditionally, autopilots achieve attitude stabilization by using rate gyros to sense and correct unwanted rotations in yaw, pitch, and roll [1]. While this method has proven to be effective and is a standard feature of many autopilot systems, it is susceptible to drift during flights over long durations. The reason is that the rate gyros only sense angular velocities, and do not provide an absolute orientation reference. Thus, the orientation of the aircraft in yaw, pitch, and roll is obtained by integrating the rate signals – a process that can lead to substantial noise-induced drift [1]. One solution to this problem is to use sensors that provide direct information on absolute orientation. For example, 3-axis magnetometers, used together with gyroscopes and/or accelerometers, can deliver information about an aircraft's absolute orientation [2]. However, when used by themselves they will not sense rotations about an axis that is parallel to the direction of the local magnetic field [2]. Another way of deriving absolute orientation is to take advantage of the fact that the sky is usually brighter than the ground (in the visible spectrum) or darker than the ground (in the infrared), and use this to determine which direction is "up". Infrared sensors are a compact, lightweight, and cost-effective means of implementing this technique to stabilize roll and pitch [3], [4]. However, they are susceptible to errors when the sun is low in the sky. Another vision-based method for determining attitude involves capturing a wide-angle view of the environment, including the horizon. The position and orientation of the horizon, obtained after segmenting the image into sky and ground, can be used to infer the aircraft's attitude [5], [6]. While this technique is more robust to variations in the sun's position, it can be computationally intensive and challenging to implement in real time because sophisticated spectral and intensity analysis is required to achieve reliable sky/ground segmentation. Another approach has been to use specially designed VLSI chips to extract horizon information in real time from an array of photodiodes [7]. While this approach is attractive, its current accuracy is likely to be limited because it uses only intensity information to locate the horizon.

Here we describe a technique for estimating the roll and pitch attitude of an aircraft, based on sensing the horizon in a visual image. The advantages of the technique are that (a) it locates the horizon reliably over a large variety of images, (b) it is relatively simple to implement in real time, (c) it can be incorporated into any vision system (e.g. narrow angle, wide angle or panoramic), irrespective of the way in which the environment is imaged (e.g. through lenses or mirrors), and (d) it can be piggybacked onto an existing vision system that serves other functions, such as terrain following, obstacle avoidance, and surveillance.

The goal of this work is to produce a robust horizon detection method that is computationally efficient, allowing implementation in real-time while leaving time for other image processing tasks.

II. Horizon Detection Method

The proposed horizon detection method consists of four stages:

A. Enhance sky/ground contrast
B. Determine optimum threshold for sky/ground segmentation
C. Convert horizon points to vectors in the viewsphere
D. Fit 3D plane to horizon vectors to estimate attitude

A. Enhance Sky/Ground Contrast

Arguably, the most important stage of visual horizon detection is the accurate separation of sky from ground. This problem has been tackled in various ways including morphological smoothing [8], using the blue image channel with or without intensity weighting [9], using a support vector machine (SVM) [10], using a combination of color, texture and a Hidden Markov Tree [11], Multiscale Linear Discriminant Analysis (MLDA) [12], and interclass difference maximization [13].

Manuscript received March 2, 2009. This work was supported partly by US Army Research Office MURI ARMY-W911NF041076, Technical Monitor Dr Tom Doligalski, US ONR Award N00014-04-1-0334, ARC Centre of Excellence Grant CE0561903, and a Queensland Smart State Premier's Fellowship.

Authors are associated with the Queensland Brain Institute and the School of Information Technology and Electrical Engineering, University of Queensland, St Lucia, QLD 4072, Australia and also ARC Centre of Excellence in Vision Science, Australia (email: s.thurrowgood@uq.edu.au).

In this study we seek a single linear transformation of the RGB color space that provides, with high probability, a good separation of sky and ground classes. Because many approaches have used image brightness or a single color channel as the input for classification, we wish to investigate and quantify the benefit of using a combination of the three standard color channels. This dimensionality reduction is performed as a preprocessing of the image to reduce computational complexity. We used a data set of 124 images collected from flight trials, and from the internet, that provided large variations in the color, brightness and texture of sky and ground e.g., snow, desert, farm land, and inner city, as well as variations in the white balance and the cameras that were used to capture the images. Each image was segmented into sky and ground by a human operator (S.T.) to obtain "ground truth" data.

Methods that depend on brightness alone will sample approximately the *Y* component of the *YUV* color space, and make the reasonable assumption that sky is brighter than ground. The conversion from red, green, and blue (*RGB*) image channels into brightness (*Y*) and color (*UV*) is given in (1), where *RGB* values span the range [0,255].

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0.3 & 0.586 & 0.114 \\ -0.168 & -0.332 & 0.5 \\ 0.5 & -0.418 & -0.082 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (1)$$

The histogram of color (*UV*) components for all pixels in our data set images is shown in Fig. 1a. We see that, in this transformation, points representing the sky (blue) lie approximately in the opposite direction to points representing the ground (green/brown). The color component can effectively be reduced from two dimensions down to one by rotating the *UV* space such that the majority of the data is aligned with the *U'* axis as shown in Fig. 1b.

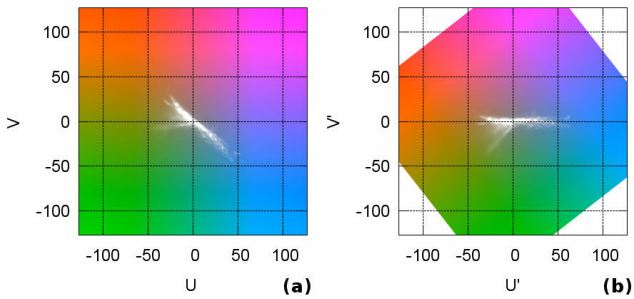

Fig. 1. (a) Histogram of the *UV* color components of every pixel of all 124 images. Brighter whites contain more points. The background depicts the colors of points in this color space. (b) Same as (a) but rotated by 38 degrees to new axes *U'V'*.

Using the manually segmented data set we can plot the histograms of brightness and transformed color separately for sky and ground, as in Fig. 2 and 3. It can be seen in these figures that, on average, there is a large overlap in both the brightness and the color of sky and ground.

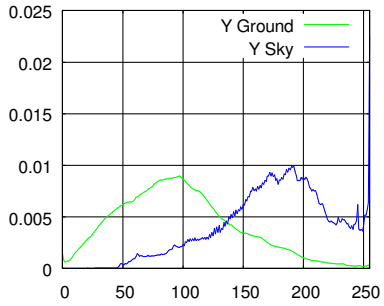

Fig. 2. Histogram of pixel brightness (*Y*) plotted separately for sky and ground. Each curve is normalized by the total number of sky or ground pixels respectively. Classes overlap by 30.5%.

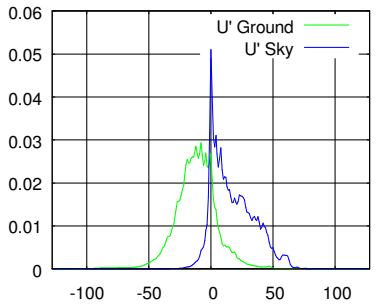

Fig. 3. Histogram of transformed pixel color (*U'*) plotted separately for sky and ground. Each curve is normalized by the total number of sky or ground pixels respectively. Classes overlap by 22.2%.

If we were to segment our images by applying a threshold, then the best choice of threshold would be at the location where the sky/ground histograms cross. From these plots we see that such a threshold performed on either brightness or color alone would result in a significant percentage of misclassified pixels, 30.5% if using *Y*, and 22.2% for *U'*.

Further, we plot in Fig. 4 the histogram of brightness versus color. We see in this plot that the sky and ground classes can be further distinguished by combining information from the two histograms appropriately.

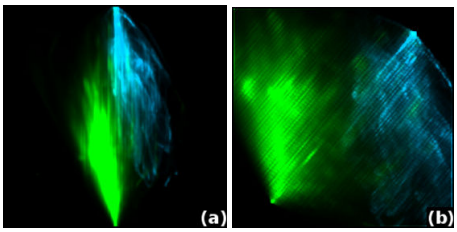

Fig. 4. 2D Histograms of color (*U'*, horizontal axis) versus brightness (*Y*, vertical axis). The blob in the lower-left represents the ground pixels while the blob in the upper right represents the sky pixels from manual classification. (a) Histogram of *U'* versus *Y*, (b) histogram of 4*U'* versus *Y* and rotated by 40 degrees.

If we use only the horizontal axis of the brightness/color histogram in Fig. 4b by projecting the vertical axis onto the horizontal, which we will call *C*, we obtain a substantial reduction in the overlap of the sky/ground distributions down to 8.6%, as shown in Fig. 5. This shows that we can get a better separation of sky and ground by combining the statistics of brightness and color. The transformation from

RGB to *C* is given in (2) which is obtained by performing the stated transformations on the *YUV* color space. Note that at runtime, the only computation performed for sky/ground contrast enhancement is the use of (2).

$$C = -1.16R + 0.363G + 1.43B - 82.3 \quad (2)$$

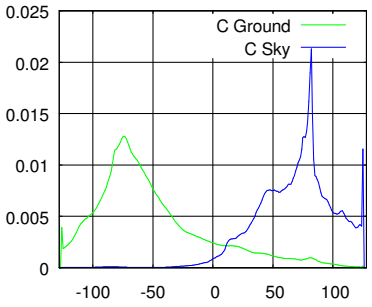

Fig. 5. Histogram of combined pixel brightness and color (*C*) plotted separately for sky and ground. Each curve is normalized by the total number of sky or ground pixels respectively. Classes overlap by 8.6%.

The above procedure attempts to find a projection onto a single line through the *RGB* color space that minimizes the overlap of the sky/ground classes. This can also be accomplished through classical linear discriminant analysis methods such as Fisher's Linear Discriminant (FLD) [14]. Methods such as Ettinger's [13] use something akin to FLD to exhaustively search for the maximum difference between sky/ground classes in each image, which can be very computationally expensive. On the other hand, preprocessing the image with an optimal linear transformation is computationally efficient and effective in reducing the sky/ground overlap to less than 9%.

B. Determine Optimum Threshold for Sky/Ground Segmentation

We will now be working with histograms as they are measured from images, without foreknowledge to which class each pixel belongs, which gives a histogram similar to the sum of the sky/ground histograms in Fig. 5. In practice we divide *C* by two to avoid the saturation seen in Fig. 5, and we add 128 and clamp *C* to the range [0,255].

It is intuitively clear that the crossover point of the sky/ground distributions will change from one image to the next. Such changes can occur with changes in the relative areas of sky/ground and also in images where the ground is unusually similar, or unusually different, from the sky. This can be caused by variations in white balance, or flight at very high altitudes resulting in a blue haze over the ground, or flight during periods of dark cloud cover.

Much previous work has been done on automatic thresholding (surveyed in [15]), but here we propose a computationally simple approach that allows prior knowledge of the average aircraft dynamics and imaging areas to be incorporated. The technique is histogram based, and relies partly on Bayesian prior probabilities. It makes use of prior knowledge in two ways. Firstly, in an imaging system in which the image captures almost the full viewsphere, we can say that the most likely location of the threshold in the integral histogram space is at the 50% point, i.e. there are likely to be just as many sky pixels as ground pixels in the image (in the special case of an imaging system that captures the full view sphere and is being flown at high altitude, sky and ground will always cover equal areas, irrespective of the orientation of the aircraft). Thus, from this standpoint, the prior probability density function for the location of the threshold can be assumed to be a Gaussian function with its peak at the 50% point (see Fig. 6b). Note that in Fig. 6b the tails of the Gaussian have been trimmed at the extreme ends because in practice these areas are a very poor choice for a threshold. Secondly, it is likely that most of the pixels in the image possess colors that correspond to either the ground or the sky, with very few pixels representing colors that do not belong to either class (or that cover both classes). From this standpoint, the most likely locations for the threshold would be at positions that correspond to local minima of the histogram (the lower the minim, the greater the likelihood). Thus, the prior probability density function for the location of the threshold can be assumed to be a Gaussian function with its peak at a histogram value of zero (see Fig. 6a).

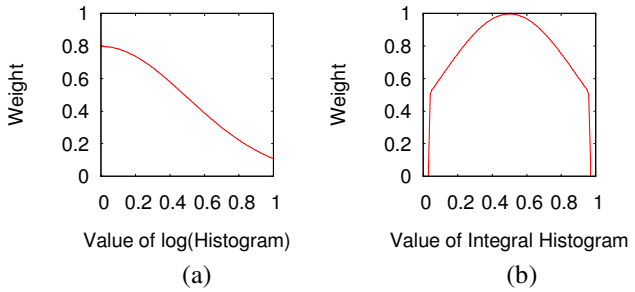

Fig. 6. Weighting functions, (a) used on the normalized logarithm of the histogram, and (b) used on the normalized integral of the histogram.

Following the above reasoning, the chosen threshold, *T* in (5), is the one with the maximum score, *S*, as calculated in (4) over all possible clamped *C* values, $c \in [0,255]$.

$$W(x) = \begin{cases} N(x,0.5,0.4) & \text{if} \quad 0.02 < x < 0.98 \\ 0.0 & \text{otherwise} \end{cases} \quad (3)$$

$$S_c = N(h_c,0,0.5) \times W(i_c) \quad (4)$$

$$T = \max(S_c) \quad (5)$$

where $N(x,\mu,\sigma)$ is the Gaussian normal distribution, h_c is the normalized log of the histogram of the image, and i_c is the normalized integral histogram.

Even though the prior Gaussian distribution for the threshold location prefers to segment the number of pixels into two equal parts, the algorithm is relatively robust even in extreme cases where this assumption is not true. The

image in the example of Fig. 8 contains less than 4% sky pixels, yet the automatic thresholding procedure chooses a threshold value of 120 for C (see Fig. 9). Without modifying the assumed width of the prior Gaussian distribution for the threshold, this case is close to the limits of the method. The best threshold value in this image would have been around 144 but the area bias has prevented the selection of such a high value. The contrast enhancement has allowed the threshold to shift without significant degradation of performance because the sky/ground classes are well separated. The performance of the method improves substantially as the percentage of sky increases to 10% or more. Examples of applying the first two stages of horizon detection to images *not* in the collected data set are provided in Fig. 7 and Fig. 8 (images from www.freephotos.lu).

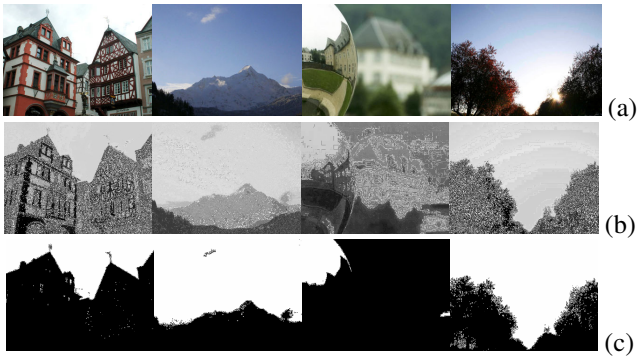

Fig. 7. Examples of (a) images *not* included in the collected data set, (b) contrast enhanced images using (2), and (c) automatic threshold results.

Fig. 8. (a) Original image with < 4% sky. (b) Contrast enhanced image. (c) Result of automatic thresholding (at a level of 120 in this case).

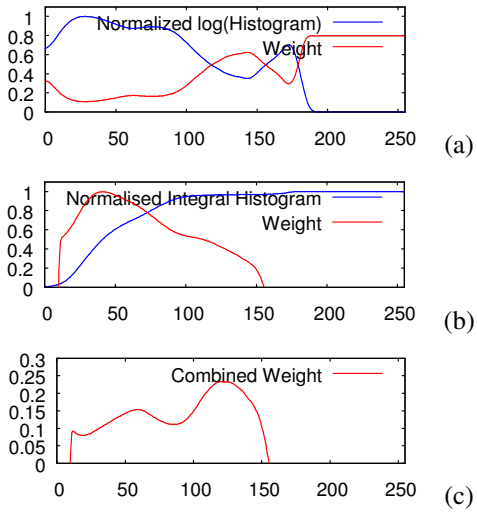

Fig. 9. Composition of the threshold score for the image in Fig. 8b. (a) Weight from the normalized log(Histogram). (b) Weight from the normalized integral histogram. (c) Combined weight which is the score S.

To check the performance of the first two stages, we ran the contrast enhancing and thresholding procedure on the entire data set of 124 images and compared this with the manual segmentations. The set contained ten very unusual images that caused relatively large misclassifications of 10% or more. However, the remaining 114 images had an average misclassification rate of 1.69% with a standard deviation of 2.38%. This is well below the expected error rate given by the overlap in the histogram of the contrast parameter C. Thus, it is clear that a substantial benefit is being gained from the automatic thresholding procedure developed here.

C. Convert Horizon Points to Vectors in the Viewsphere

Once the horizon points are determined by using the threshold, we compute the viewing direction of each of these points as a unit vector in a viewsphere centered on the nodal point of the system. As a consequence, the proposed scheme will work on any imaging system (incorporating any camera, lens and/or reflective surface) as long as the system is geometrically calibrated. The model needs only a function that converts pixel coordinates to view direction. We use the "OcamCalib" toolbox [16] and a toolbox based on the calibration procedure in [17]. For a high altitude flight where the effects of horizon topography are negligible, the correct horizon points should lie on an equatorial plane that passes through the centre of the viewsphere $[0,0,0]^T$.

D. Fit 3D Plane to Horizon Vectors to Estimate Attitude

The method used for fitting a plane is the repeated application of a linear least squares fit to a subset of the horizon vectors. In this fit we constrain the plane to pass through the centre of the viewsphere which is especially beneficial in systems with a smaller field of view. Vectors are discarded from the fit based on their distance from the currently fitted plane. The termination criterion used is when we have discarded 50% of points from the fit, but we could also use a criterion based on the variance of the fitted points from the plane estimate, but at the expense of increased and more variable execution time.

III. In-Flight Performance

Flights were performed using a trainer aircraft with various vision systems mounted to the front. The engine was repositioned above the wings to give the vision system a less obscured forward view. A Microstrain inertial measurement system was mounted rigidly to the vision system to provide an attitude reference.

The plots in Fig. 11 are from a manually controlled flight using a vision system that is specially designed for terrain following [18]. Fig. 10 shows one image from this flight. We measured the error statistics between the horizon-based attitude estimates described here, and inertial estimates. The standard deviations of the differences in roll and pitch were 3.31 and 3.01 degrees respectively.

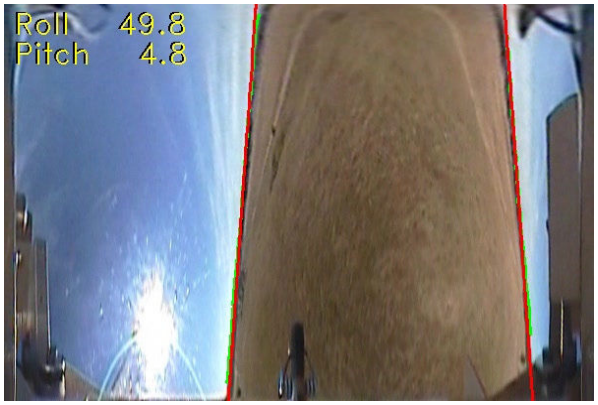

Fig. 10. Image from an in-flight trial at time=102.3s. Green points indicate the boundary between sky/ground. Red line is the fitted horizon plane. Note fewer horizon outliers at altitude.

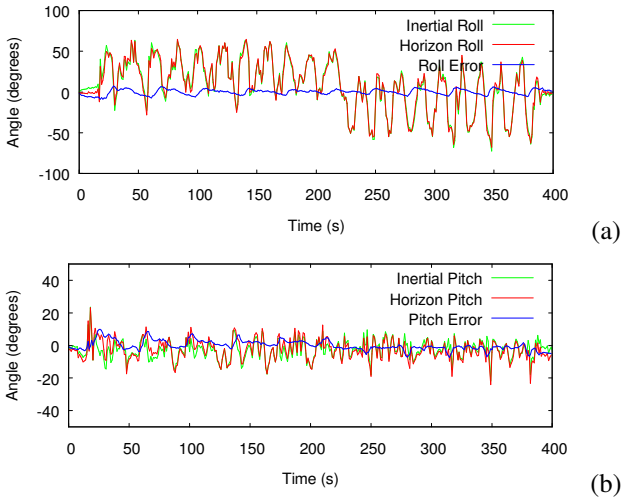

Fig. 11. (a) Horizon roll estimate, and (b) horizon pitch estimate compared to inertial roll and pitch estimates during an in-flight trial.

We note that there is a periodic variation in the roll and pitch errors. To determine if this variation was due to the inertial estimate we compare this to a hand-held test, with results shown in Fig. 13, and a representative image from the sequence in Fig. 12. The hand-held sequence used a forward looking fisheye lens with a 180 degree field of view. The periodic component of the error was very small in this case. So we conclude that the majority of the error in the flying sequence was due to a large periodic error in the inertial estimate during flight. The standard deviations of roll and pitch error for the hand-held sequence were 1.39 and 1.69 degrees respectively.

Upon inspecting the video sequences, one might expect the hand-held sequence to have a higher error variance than the in-flight sequence because the trees at ground level can be seen to cause noise in the attitude estimate. Such errors should decrease with altitude. However, we observed the opposite, probably due to the periodic variation in the inertial signal. So, we expect the true error variance in flight to actually be less than that measured on the ground.

Short segments of these two test sequences are contained in the accompanying video.

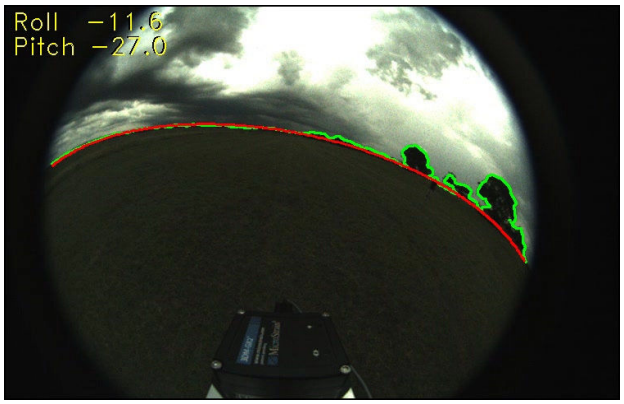

Fig. 12. Image from the hand-held sequence at time=63.7s. Green points indicate the boundary between sky/ground. Red line is the fitted horizon plane. Note dark clouds ignored by thresholding, trees ignored by plane-fit.

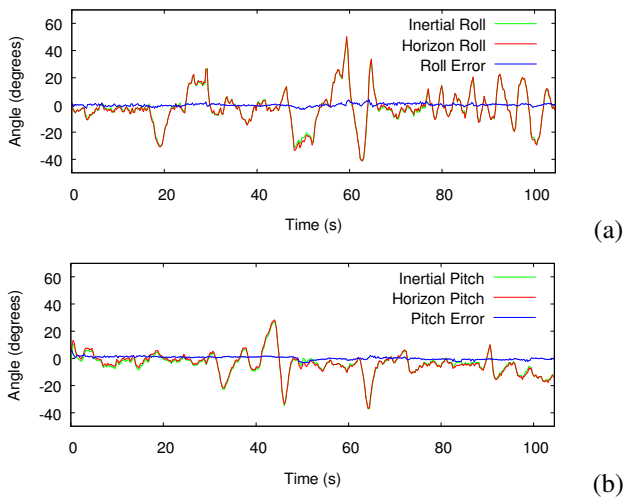

Fig. 13. (a) Horizon roll estimate, and (b) horizon pitch estimate compared to inertial roll and pitch estimates during a hand-held trial.

IV. Complexity Comparison with Other Methods

Due to the difficulty in obtaining ground-truth for aircraft attitude, most papers do not provide a quantitative measure of error in their estimates of roll and pitch. However, we have included in Table I a comparison of execution times for various published studies on visual attitude estimation.

TABLE I
Comparison of Execution Times For Visual Attitude Estimation

Estimation Method	Run Time at 1.0GHz (ms)	Notes
Proposed method	2.4	300x200, 1.2ms @ 2.0GHz
Inter-class difference maximization, [13], used in [7] and [19].	30	320x240, 33ms @ 0.9GHz
MLDA, [12]	80	128x128, 33ms @ 2.4GHz
Wavelet descriptors, [11]	216	640x480, 120ms @ 1.8GHz
SVM segmentation + Hough transform, [10]	224 to 504	320x240, 120ms + 200 to 600ms @ 0.7GHz
Morphological smoothing + Hough transform, [8]	300	352x288, 67ms @ 3.0GHz

V. Performance Comparison with Other Methods

We have implemented the method in [13] to compare to our own flight sequences and found that both methods produce almost identical results, with the exception of extreme noise. Because we robustly fit a plane to horizon candidate points we can effectively ignore around 50% of extreme outliers, while a global search, as described in [13], will often be offset. That being said, the method in [13] is superior when the noise is image noise rather than misclassification, such as when using analog video transmission, but our system is implemented onboard the aircraft so there is no such extreme noise and thus no performance cost incurred.

Compared to others, the proposed method is very efficient to execute due to the prior fitting of a model that allows dimensionality reduction of the image data. This approach is very different to [13] where an exhaustive search over all values of roll/pitch is performed in every frame to find a maximum sky/ground class difference. Our efficiency comes at the cost of having no guarantee that it will operate accurately with scenes that are significantly different from the training data e.g. operation over oceans is unlikely to be successful with our current training data.

From our tests we have seen that our method performs well in varied conditions of cloud, sun, altitude and different imaging systems, and is resistant to false horizon edges e.g. from large trees, sky reflection from lakes or from CCD camera blooming when the sun is in full view.

VI. Conclusion

This paper has introduced a robust and computationally efficient method for visual attitude estimation, useful in control of UAVs. We have quantified the misclassification rate in sky/ground classes when using brightness only, color only, and a near optimal combination of both. The proposed method makes best use of color information for the separation of sky/ground classes, but will also work on brightness images with a significantly higher rate of pixel misclassification, due to the error-correcting attributes of the robust plane fitting.

Compared to other methods, we have used a simple and fast, yet effective method for automatic threshold selection that is successful in reducing the expected misclassification rate.

We introduce the idea of fitting the horizon in a camera independent way, by using a camera calibration model to convert horizon points into viewsphere vectors. This allows the horizon detection system to be piggybacked onto any existing imaging system whose geometry has been calibrated.

The combination of techniques used in this work form a robust horizon detection system that executes in a relatively short time and has the potential to be implemented very efficiently. Future work will include gathering statistics for the priors used in thresholding, and automating the color space optimization.

Acknowledgment

We would like to thank David Brennan for kindly allowing us to use his private air field for flight tests.

References

[1] J. Rohac, "Accelerometers and an aircraft attitude evaluation," *Proceedings IEEE Sensors Conference*, Nov. 2005.

[2] S. Merhav, *Aerospace Sensor Systems and Applications*. Springer, 1996, pp. 415-426.

[3] A. Jalink, Jr., R. E. Davis, and J. A. Dodgen, "Conceptual design and analysis of an infrared horizon sensor with compensation for atmospheric variability," NASA Technical Note TN-D-6616, Feb. 1972.

[4] J.A. Gozdecki, "Aircraft attitude sensor and feedback control system," United States Patent, Number 6181989, Jan. 2001.

[5] T. D. Cornall, A. Price, and G. K. Egan, "Measuring horizon angle on a small unmanned air vehicle using digital video camera and an FPGA," *In Proc. ICARA 2006, International Conference on Autonomous Robots an Agents*, Dec. 2006.

[6] T.D. Cornall, G.K. Egan, and A. Price, "Aircraft attitude estimation from horizon video," *IEE Electronics Letters*, 42(13), June 2006, pp. 744-745.

[7] T.K. Horiuchi, "A low-power visual horizon estimation chip," *Proceedings, IEEE International Symposium on Circuits and Systems* (ISCAS), May 2005. pp 4755-4758

[8] D. Dusha, W. Boles, and R. Walker, "Fixed-wing attitude estimation using computer vision based horizon detection," *In Proceedings 12th Australian International Aerospace Congress*, April 2007.

[9] T. Cornall, "UAV attitude from horizon video," PhD Thesis, Monash University, Dec. 2006.

[10] T. G. McGee, R. Sengupta, and K. Hedrick, "Obstacle detection for small autonomous aircraft using sky segmentation," *Proceedings of the IEEE International Conference on Robotics and Austomation*, April 2005.

[11] S. Todorovic, M. C. Nechyba, and P. G. Ifju, "Sky/ground modeling for autonomous MAV flight," *Proceedings IEEE International Conference on Robotics and Automation* (ICRA), 2003, vol.1 pp. 1422-1427.

[12] S. Todorovic, and M. C. Nechyba, "A vision system for intelligent mission profiles of micro air vehicles," *IEEE Trans.on Vehicular Technology*, 2004, pp. 1713-1725.

[13] S. M. Ettinger, M. C. Nechyba, P. G. Ifgu, and M. Waszak, "Vision-guided flight stability and control for micro air vehicles," *Proceedings IEEE International Conference on Intelligent Robots and Systems*, 2002, vol.3 2134- 2140.

[14] R. Duda, P. Hart, and D. Stork, *Pattern Classification*. Wiley, 2000.

[15] M. Sezgin and B. Sankur, "Survey over image thresholding techniques and quantitative performance evaluation," *Journal of Electronic Imaging*, Vol.13, No.1. 2004, pp. 146-168.

[16] D. Scaramuzza, A. Martinelli, and R. Siegwart, "A toolbox for easily calibrating omnidirectional cameras," *Proceedings IEEE International Conference on Intelligent Robots and Systems*, 2006.

[17] J. Kannala and S. S. Brandt, "A generic camera model and calibration method for conventional, wide-angle, and fish-eye lenses," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2006.

[18] D. Soccol, S. Thurrowgood, and M. V. Srinivasan, "A vision system for optic-flow-based guidance of UAVs," Proceedings, Ninth Australasian Conference on Robotics and Automation, 2007.

[19] C. De Watger, B. Bijnens, and J. A. Mulder, "Vision-only control of a flapping MAV on Mars," *AIAA Guidance, Navigation and Control Conference*, 2007.